Electronic Version
Stylesheet Version v1.1.1

Description

ACCURATE TIMING ANALYSIS OF INTEGRATED CIRCUITS WHEN COMBINATORIAL LOGIC OFFERS A LOAD

BACKGROUND OF INVENTION

[0001] *Field of the Invention*

[0002] The present invention relates to computer-aided design (CAD) of integrated circuits, and more specifically to a method and apparatus to perform accurate timing analysis of integrated circuits when combinatorial logic offers a load to other components of an IC

[0003] *Related Art*

[0004] Integrated circuits (ICs) generally contain various components (flip-flops, pins, logic gates, multiplexers, etc) connected in a desired topology. Components such as logic gates and multiplexers, which do not store bit values or state information in general, are generally referred to as combinatorial logic. On the other hand, components such

as flip-flops, which store state information, are referred to as sequential elements.

[0005] Timing analysis is often performed during the design phase of an IC. Typically, digital data representing the IC design is analyzed using various design tools. Timing analysis is generally performed to ensure that the signals are propagated by combinatorial logic and captured by sequential elements in a time duration permitted by the period of a clock signal used to drive the IC. In general, it is desirable to use a clock signal of short period to the extent timing violations are not experienced in the operation of an IC, and timing analysis is performed to detect any potential timing violations.

[0006] One factor considered in timing analysis is the load offered at the output of each component ("present component") by a subsequent component (connected to the output). A higher load (offered by the subsequent component(s)) generally requires greater propagation time for a signal generated (as an output) by the present component. In general, the propagation times need to be considered in timing analysis since the clock frequency/period driving an IC is determined by the propagation times.

[0007] At least for such a reason it is desirable to perform timing

analysis, which takes into consideration accurate load offered by combinatorial logic.

Brief Description of Drawings

[0008] The present invention will be described with reference to the following accompanying drawings.

[0009] Figure (Fig.)1 is a diagram illustrating the details of an example circuit in which various aspects of the present invention may be implemented.

[0010] Figure 2 is a timing diagram illustrating the manner in which setup and hold violations may be encountered if the capacitive load of a combinatorial element is not accurately computed.

[0011] Figure 3A is a flowchart illustrating the manner in which accuracy of timing analysis may be improved according to an aspect of the present invention.

[0012] Figure 3B is a circuit diagram depicting the details of a cell containing multiple logic gates.

[0013] Figures 4A-4D illustrate the manner in which the capacitance may be computed and the approximate capacitance value when the output of a NAND gate switches.

[0014] Figures 5A-5D illustrate the manner in which the capacitance may be computed and the approximate capacitance value when the output of a NAND gate does not switch.

[0015] Figure 6 is a block diagram illustrating the details of a system operable by software instructions to provide several features of the present invention.

[0016] In the drawings, like reference numbers generally indicate identical, functionally similar, and/or structurally similar elements. The drawing in which an element first appears is indicated by the leftmost digit(s) in the corresponding reference number.

DETAILED DESCRIPTION

[0017] *1. Overview*

[0018] An aspect of the present invention improves the accuracy of timing analysis of an integrated circuit based on a recognition that the capacitance (and thus the load) offered at an input pin of a combinatorial element is higher when the output of the combinatorial element switches state compared to a scenario in which the output does not change. A combinatorial element generally refers to components such as logic gates.

[0019] In an embodiment described below, a first capacitance value offered at an input pin of a combinatorial element is determined when the output path (of the combinatorial element) is switching. Similarly, a second capacitance

value at the input pin is determined when the output path is not switching. The first capacitance value is associated to the input pin of the combinatorial element while performing timing analysis of the integrated circuit for setup time errors. Similarly, the second capacitance value is associated to the input pin of the combinatorial element while performing timing analysis of the integrated circuit for hold time errors.

[0020] Such a determination and association of different value of capacitance to the input pin of the combinatorial element for different analysis conditions generally results in accurate representation of propagation delays in the operation of the combinatorial logic and hence may result in accurate timing analysis.

[0021] Several aspects of the invention are described below with reference to examples for illustration. It should be understood that numerous specific details, relationships, and methods are set forth to provide a full understanding of the invention. One skilled in the relevant art, however, will readily recognize that the invention can be practiced without one or more of the specific details, or with other methods, etc. In other instances, well structures or operations are not shown in detail to avoid obscuring the in-

vention.

[0022] *2. Example Circuit*

[0023] Figure 1 is a circuit diagram of an example circuit, which is used to illustrate the manner in which timing analysis of ICs can be performed accurately according to several aspects of the present invention. The example circuit of Figure 1 is shown containing flip-flops 110 and 160, NAND gate 120 and inverter 130. Each element is described below in further detail.

[0024] The output (path 112) of flip-flop 110 is shown connected to pin B of NAND gate 120 and pin C of inverter 130 and hence the load on output path 112 is affected by capacitance at pin B of NAND gate and capacitance at pin C of inverter 130. When a higher load is offered on path 112, flip-flop 110 may generate output signals with a corresponding delay.

[0025] Flip-flop 160 is shown clocked by CLK signal and receiving the output of inverter 130 on D-input. As may be appreciated, for accurate latching of data received on D-input, the D-input and CLK input need to be received with appropriate timing relationship (as described in further detail below with reference to Figure 2). Over-estimation or under-estimation of delays in the generation of output

signals of flip-flop 110 (and inverter 130) may lead to inaccurate operation of flip-flop 160.

[0026] NAND gate 120 represents an example combinatorial element, which offers a higher load at pin B (path 112) when the output at path 122 switches, in comparison to a situation in which path 122 does not switch. As may be appreciated, the output of a 2-input NAND gate switches in response to a bit change on one pin, when the other pin is at logic high (1). As may be readily observed, NAND gate 120 is driven by flip-flop 160.

[0027] An aspect of the present invention takes advantage of the observation that the offered load is different depending on whether the output of a combinatorial element switches or not, and enables accurate timing analysis of integrated circuits, as described below with examples. Example problems that would be presented if difference of load is not taken into account, is first described below.

[0028] *3. Example Problems*

[0029] Figure 2 is a timing diagram illustrating the example problems caused if the load offered on an output path is underestimated or overestimated. Waveforms 210 and 230 illustrate the desired timing relationship for proper operation of flip-flop 160. In particular, waveforms 210

and 230 respectively represent the CLK and D-inputs of D-flip flop 160.

[0030] The timing of waveforms 210 and 230 needs to be such that duration (236) between rising edge 235 of D input and clock edge 215 needs to be greater than or equal to a setup time required by the implementation of D-flip flop. Similarly, the duration (237) between clock edge 215 and falling edge 238 of D input also needs to be greater than or equal to a hold time specification.

[0031] Waveform 250 illustrates the failure of setup time requirement if the load offered by NAND gate 120 on path 112 is substantially more than an expected value. As may be appreciated, in case of such underestimation, D-input would be received later than expected. Thus, the D-input of flip-flop 160 would be received later than waveform 230, as depicted by waveform 250. Assuming that time duration 256 between clock edge 215 and rising edge 255 (of waveform 250) is less than the corresponding setup time,OLE_LINK1 flip-flop 160 may not latch the data value on D-input accurately (which is undesirable).

[0032] OLE_LINK1Waveform 260 represents the D-input if the load offered by NAND gate 120 on path 112 is substantially overestimated. In comparison to waveform 250,

waveform 260 occurs sooner in time. Assuming that time duration 267 between clock edge 215 and falling edge 268 (of waveform 260) is less than the corresponding hold time, flip-flop 160 may not latch the data value on D-input accurately (which is undesirable).

[0033] While the circuit of Figure 1 merely contains a few representative elements for conciseness, it should be understood that the errors in timing analysis may be compounded with many more elements being connected in series, and thereby eventually leading to the setup time and hold time violations somewhere in the integrated circuits during operation (production/application deployment).

[0034] It is therefore desirable that the timing analysis takes into account the difference of offered loads (by combinatorial elements). A cell library may be accurately characterized using various aspects of the present invention, as described below in further detail with reference to Figure 3A.

[0035] *4. Characterizing Cell Libraries for Accurate Timing Analysis*

[0036] Figure 3A is a flow chart illustrating the manner in which cells (formed of combinatorial logic) contained in each cell library can be characterized for accurate timing analysis according to various aspects of the present invention. As

is well known, multiple libraries are provided for the same functional blocks, with each library generally being designed for a specific combination of PVT (process, voltage, temperature) conditions.

[0037] In addition, each cell may be designed using one or more logic gates or custom designed. A combinatorial element refers to each of such logic gates or a portion of the custom circuit which behaves as a logic gate in terms of switching the value on a path (either internal to the cell or providing as an external output) when appropriate input vectors are applied.

[0038] Continuing with reference to Figure 3A, the method of Figure 3 is described with reference to example circuit of Figure 1 for illustration. The flow chart begins at step 301 and control is immediately passed to step 310.

[0039] In step 310, a first set of input vectors that would cause the combinatorial element connected to the pin of interest to switch. In the example circuit of NAND gate 120 of Figure 1, the input vectors may equal 10 (1 on pin A and 0 on pin B) and 11 assuming pin B is sought to be characterized.

[0040] In step 320, a second set of input vectors that would not cause the stage connected to the pin of interest may be

determined. With respect to NAND gate 120, the input vectors may equal 00 and 01.

[0041] In step 330, a determination is made as to whether a cell of interest needs to be characterized to check setup time or hold time violations. In an embodiment, if the cell is from a library designed for a strong process corner (and/or high temperatures), the signals in the data path are assumed to be propagating faster, and thus hold time violation is of particular concern. On the other hand, if the cell is from a library intended for a weak process corner, setup time violation is of particular concern. Using such considerations, a determination may be made as to whether the cell needs to be characterized for setup time or hold time violation. The data in a library may provide the corresponding information (as to whether the library is designed for strong or weak processor corner, etc.).

[0042] In step 340, the capacitance of the pin(s) of interest is measured when the second set of input vectors are applied to the combinatorial element. With reference to the example circuit of figure 1, system may determine a capacitance value at pin B when the second set of input vectors noted above are applied to the NAND Gate 120. In general, design tools (such as SPICE simulator or related

utilities) provide the ability to measure the capacitance, and any of such tools may be used to measure the capacitance. The principle underlying an example approach is described below with reference to Figures 4A-4D and 5A-5D. Control then passes to step 360.

[0043] In step 350, the capacitance of the pin of interest is measured when the first set of input vectors are applied to the stage. The capacitance may be measured similar to as in step 340. The capacitance thus measured may represent a maximum capacitance (Cmax), and thus maximum load offered by NAND gate 120. Control then passes to step 360.

[0044] In step 360, the capacitance thus measured may be associated to the pin in the library. In general, such association is performed by storing the corresponding value in a pre-determined location/data structure, depending on the implementation of the design tools. In step 370, timing analysis of the integrated circuit is performed taking into consideration the capacitance values thus stored in step 360.

[0045] Due to the use of one of the two capacitance values as described above, the actual capacitance offered at each pin during production/application use may equal the val-

ues associates in step 360. Accordingly, the timing analysis may be accurate. The flow chart ends at step 399.

[0046] Even though the description above is provided with reference to a cell containing a single logic gate, it should be appreciated that the technique can be applied to cells containing multiple logic gates as well. In such a scenario, the behavior/operation of the combinatorial element directly connected to the input pin sought to be characterized is of particular interest. For example, with reference to the cell of Figure 3B containing NAND gate 380 and OR gate 390, to characterize pin D, input vectors are determined with reference to pins D and E such that the value on path 389 switches (step 310). The description is continued with an illustration of determination of capacitance values at pin B can be determined.

[0047] *5. Determining Capacitance Values*

[0048] Figures 4A-4D and 5A-5D contain waveforms illustrating the manner in which the capacitance at an input pin can be determined and also that the capacitance in case an output path switches is greater than in the case when the output path does not switch. In particular, Figures 4A through 4D are related to the case in which output path is switching, and Figures 5A through 5D are related to the

case in which output path is not switching. In all the 8 timing diagrams, the X-axis represents time scale and Y-axis will be clear from the context.

[0049] With reference to Figures 4A-4D, assuming pin B of Figure 1 is sought to be characterized, pin A is maintained at 1, and an input transition (e.g., 0 to 1) is applied at pin B. Figure 4A depicts the voltage level of pin B rising from logic 0 (0 volts) to logic 1 (1.5 Volts). Figure 4B depicts the output value on path 122 switching from logic high to logic low. Figure 4C depicts the current on input path corresponding to pin B. As may be easily observed, there is a high current (of the order of -8 micro ampere maximum). The charge at pin B may be computed by integrating the curve of Figure 4C, and the resulting values are represented by the curve of Figure 4D.

[0050] With reference to Figures 5A-5D, pin A is maintained at 0, and an input transition (e.g., 0 to 1) is applied at pin B. Figure 5A depicts the voltage level of pin B rising from logic 0 (0 volts) to logic 1 (1.5 Volts). Figure 5B depicts the output value on path 122 remaining unchanged. Figure 5C depicts the current on input path corresponding to pin B. As may be easily observed, there is a lower amount of current (of the order of -2 micro ampere maximum).

The charge at pin B may be computed by integrating the curve of Figure 5C, and the resulting values are represented by the curve of Figure 5D.

[0051] The capacitance value at any point may be determined by dividing the corresponding charge value on curves of Figures 4D and 5D by the voltage level. In one embodiment, the charge at time points 450 of Figure 4D and 550 of Figure 5D respectively equal 4.9819 fCoulombs and 3.0544 fCoulombs. By using such different values for difference libraries as described above, the timing analysis may be performed accurately.

[0052] While the description above is provided with reference to characterizing cell libraries, it should be appreciated that the approaches can be used in general for determining whether an integrated circuit can be operated at a desired clock speed. The load offered with and without switching is considered in performing timing analysis. The description is continued with a illustration of implementing various aspects of the present invention as a software program.

[0053] *6. Software Implementation*

[0054] Figure 6 is a block diagram illustrating the details of computer system 600 in an embodiment of the present inven-

tion. Even though computer system 600 is described with specific components and architecture for illustration, it should be understood that the present invention might be implemented in several other types of embodiments.

[0055] Computer system 600 may contain one or more processors such as central processing unit (CPU) 610, random access memory (RAM) 620, secondary memory 630, graphics controller 660, network interface 680, and input interface 690. All the components except display unit 670 may communicate with each other over communication path 650, which may contain several buses as is well known in the relevant arts. The components of Figure 6 are described below in further detail.

[0056] CPU 610 may execute instructions stored in RAM 620 to provide several features of the present invention. CPU 610 may contain multiple processing units, with each processing unit potentially being designed for a specific task. Alternatively, CPU 610 may contain only a single processing unit. RAM 620 may receive instructions from secondary memory 630 using communication path 650. Data representing cell libraries and associated data values computed above may be stored and retrieved from secondary memory 630 (and/or RAM 620) during the execution of the in-

structions.

[0057] Graphics controller 660 generates display signals (e.g., in RGB format) to display unit 670 based on data/instructions received from CPU 610. Display unit 670 contains a display screen to display results from timing analysis. Input interface 690 may correspond to a keyand/or mouse, and generally enables a user to interact and provide inputs while performing timing analysis.

[0058] Secondary memory 630 may contain hard drive 635, flash memory 636 and removable storage drive 637. Secondary storage 630 may store the software instructions and data, which enable computer system 600 to provide several features in accordance with the present invention. Some or all of the data and instructions may be provided on removable storage unit 640, and the data and instructions may be read and provided by removable storage drive 637 to CPU 610. Floppy drive, magnetic tape drive, CDdrive, DVD Drive, Flash memory, removable memory chip (PCMCIA Card, EPROM) are examples of such removable storage drive 637.

[0059] Removable storage unit 640 may be implemented using medium and storage format compatible with removable storage drive 637 such that removable storage drive 637

can read the data and instructions. Thus, removable storage unit 640 includes a computer readable storage medium having stored therein computer software and/or data. An embodiment of the present invention is implemented using software running (that is, executing) in computer system 600.

[0060] In this document, the term A computer program product@ is used to generally refer to removable storage unit 640 or hard disk installed in hard drive 635. These computer program products are means for providing software to computer system 600. As noted above, CPU 610 may retrieve the software instructions, and execute the instructions to provide various features of the present invention described above in detail.

[0061] *7. Conclusion*

[0062] While various embodiments of the present invention have been described above, it should be understood that they have been presented by way of example only, and not limitation. Thus, the breadth and scope of the present invention should not be limited by any of the above described exemplary embodiments, but should be defined only in accordance with the following claims and their equivalents.